W e b E n g i n e e r i n g

Lab 02 **Marks 100**

Instructions

Work on this lab individually.
You can use your books, notes, handouts etc. but you are not allowed to borrow anything from your peer student.

Objectives

Today's lab will help you to refresh your programming concepts in Java.

Submission

Not required.

What you have to do

Program the following tasks in Java, compile and execute them. The name of your files will be according to the task given in this lab.

Task 1 [50]

Implement a class **Dictionary** which contains two private data members i.e., *word* and its *meaning*. Provide the complete functionality of a dictionary (up to 10 words) as describes in the following:

- Provide constructors (default, parameterize, and copy), setter/getter functions to initialize the instance variables.
- *boolean addWord (Dictionary Obj)* for adding the word in the dictionary. Returns true if the word is successfully added, false otherwise.
- *Dictionary searchWord(String word)* to search the meaning of a word in the dictionary. Returns the meaning if found, "not found" otherwise.
- *void printCompleteDict()* to print the complete maintained dictionary.
- Create **Driver** class with a menu (of all the available options) to demonstrate the execution of your program.

Task 2 [50]

Write a class **Matrix** with two different private attributes *rows* and *columns*. Provide constructors (default, parameterized and copy), and setter/getter methods to initialize the private data members. Your program should provide following functionality:

- *getRows()* to get the total number of rows.
- *getColumns()* to get the total number of columns.
- *setElement()* to set the elements of the matrix at given position (i,j)
- *Add(Matrix obj1, Matrix obj2)* to add two different matrices. If matrices cannot be added, your program should display an appropriate error message.
- *Multiply(Matrix obj1, Matrix obj2)* to multiply two different matrices. If matrices cannot multiplied, your program should display an appropriate error message.
- Create **Driver** class with a menu (of all the available options) to demonstrate the execution of your program.

☺ ☺ ☺ **BEST OF LUCK** ☺ ☺ ☺